Sharing up-to-date data with collaborators

Aug 1, 2019

Trent Newman

– Burgess Lab –

Learning objectives

At the end of this session you should be able to:

- Say what GitHub is.
- Create a free GitHub account.
- Make a new repository.
- Upload a data to your “repo”.
- Make a link from a Google Doc to your file.
- Share any changes.

Principles

Why all the fuss?

- You will have to make it again.
- Go the route that will teach you more.
- Give yourself potential for sophisticated use.

What is “Git”?

- Git is a version-control system for tracking changes in source code during software development.
- Developed in 2005 by Linus Torvalds to manage development of the open-source operating system Linux.
- It is designed for coordinating work among programmers, but it can be used to track changes in any set of files.
- The tool records line by line how files have changed and works best with text files.

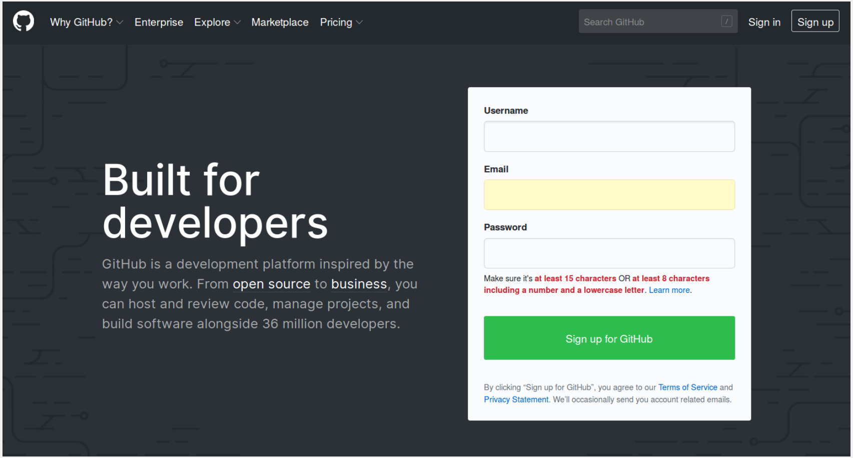
Ref: Perkel, Jeffrey. "Democratic databases: science on GitHub." Nature News 538.7623 (2016): 127.

The GitHub platform

Figure 1. Octocat – the GitHub mascot.

- GitHub provides a free, social, browser interface for the Git software.
- The site now hosts millions of projects that are kept in repositories, or "repos".
- GitHub imposes some file limitations.
  - Doesn't interface well with all file types.
  - 100 megabytes per file.
  - A gigabyte per repository.
  - Large File Storage (LFS) can allow the use of larger files.

Signing up for GitHub



Built for developers

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 36 million developers.

Username

Email

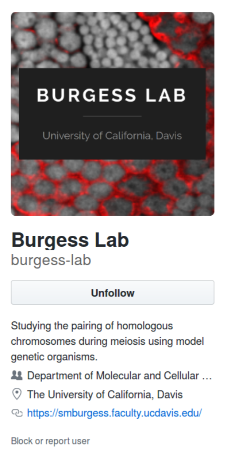
Password

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. Learn more.

By clicking "Sign up for GitHub", you agree to our Terms of Service and Privacy Statement. We'll occasionally send you account related emails.

Figure 2. The GitHub landing page.

burgess-lab



Burgess Lab
burgess-lab

Studying the pairing of homologous chromosomes during meiosis using model genetic organisms.

Department of Molecular and Cellular ...
The University of California, Davis
https://smburgess.faculty.ucdavis.edu/

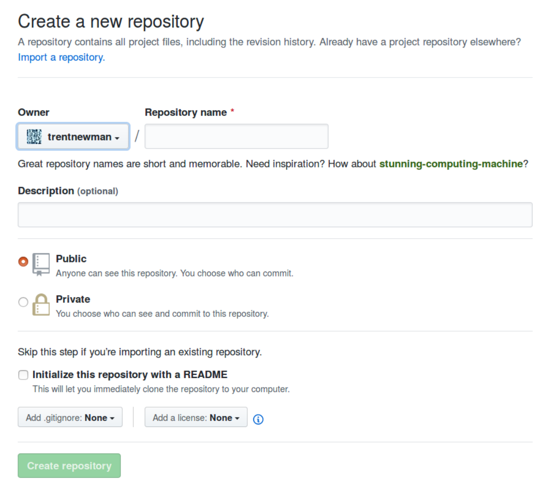
Block or report user

Figure 3. Burgess Lab GitHub.

- GitHub can be social:
  - See what projects people have worked on.
  - Follow, star, your favorite projects.
  - Provide feedback and ask questions, stay informed of updates and new features.
- Search for burgess-lab, make sure to search for **Users** and follow.

Creating a new repo

- Look for the "New repo" button, can be found on a few different pages.
- Important to decide whether the repo will be public (visible to anyone) or private.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Owner trentnewman / **Repository name** *

Great repository names are short and memorable. Need inspiration? How about **stunning-computing-machine**?

Description (optional)

● **Public**
Anyone can see this repository. You choose who can commit.

○ **Private**
You choose who can see and commit to this repository.

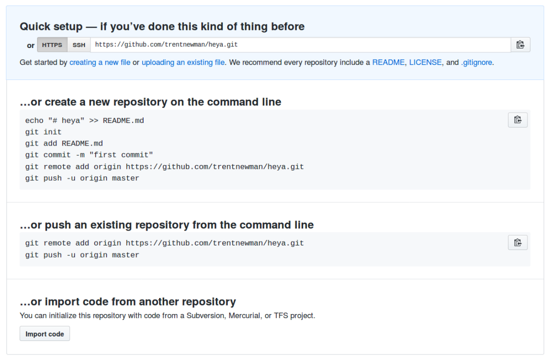
Skip this step if you're importing an existing repository.

□ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** | Add a license: **None**

Figure 4. Fill out form to create a new repository.

Initializing a repo



Quick setup — if you've done this kind of thing before

or HTTPS SSH https://github.com/trentnewman/heya.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# heya" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/trentnewman/heya.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/trentnewman/heya.git
git push -u origin master
```

...or import code from another repository

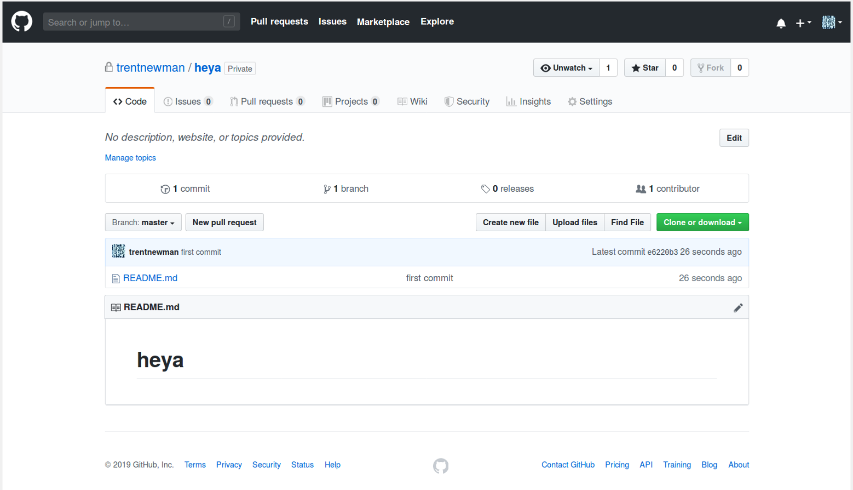
You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Figure 5. Need to upload a file to the repo to initialize it.

- The process can differ depending on whether you already have a folder that you want to make the repo out of.
- Can either create an empty repo (with a minimal file in it) or initialize an exisiting folder.

The new repo



trentnewman / heya Private

No description, website, or topics provided.

Manage topics

1 commit 1 branch 0 releases 1 contributor

trentnewman first commit Latest commit e6220b3 26 seconds ago

README.md first commit 26 seconds ago

README.md

heya

© 2019 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub Pricing API Training Blog About

Figure 6. What an empty repository looks like after being made.

Using the “command line”

It can be helpful to become familiar with the terminal which allows you to interact with your computer directly.

- On a mac the “Terminal” is located under Applications/Utilities/.
- The ls and cd commands are sufficient to navigate your files.
- To list the files in your current directory enter:

```
ls
```

- To change the directory to one of the listed folder, e.g. Documents, type:

```
cd Documents
```

- To go “up” a directory, enter:

```
cd ../
```

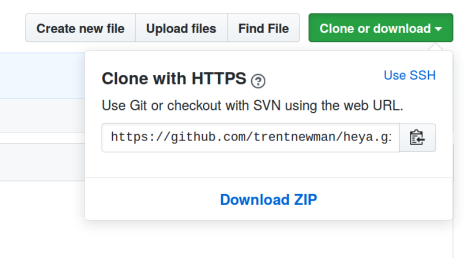
Installing Git

Git installation can be a little tricky, and varies by operating system.

- Instructions from: https://gist.github.com/derhuerst/1b15ff4652a867391f03
- For Mac, first install Homebrew.
- Run the following commands in the terminal window.

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
brew doctor
brew install git
```

Cloning a repo



Clone with HTTPS

Use Git or checkout with SVN using the web URL.

https://github.com/trentnewman/heya.g:

Figure 7. Button to copy the repo to your computer.

- Can download the whole repo as a zip, but you will want to "clone" the repo to tracking of changes.
- Open a command prompt in the directory you want to put the repo and paste the link provided, e.g.

```
git clone https://github.com/trentnewman/heya.git
```

Basic use of Git

Typically you will work with the files on the origin (your local computer) and then “push” the changes to the master (GitHub).

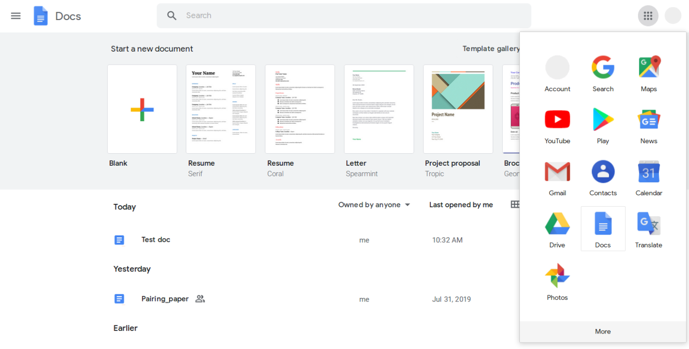
```
git status # this can tell you whether any changes have been made
git add . # add changes, ‘‘.’’ means everything in the directory
git commit -m "<enter message here>" # commit the changes, with a message describing the changes
git push origin master  # upload the committed changes from your machine to GitHub
```

Collaborative writing

- Writing a paper has traditionally involved one author sharing drafts with colleagues and then waiting for coauthors to reply with comments.
- Collaborative tools now simplify this process by allowing multiple authors to edit and format an online document at the same time.
- The most widely used general-purpose collaborative writing app is probably Google Docs.

Ref: Perkel, Jeffrey M. "Scientific writing: the online cooperative." Nature News 514.7520 (2014): 127.

Starting a new Google Doc



Today

Test doc	me	10:32 AM

Yesterday

Pairing_paper	me	Jul 31, 2019

Earlier

Figure 8. Google Docs.

- Using these tools requires a Gmail account.
- Click on the Docs icon and create a new blank document.

Collaborating on a Doc

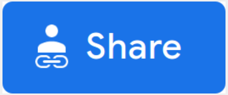
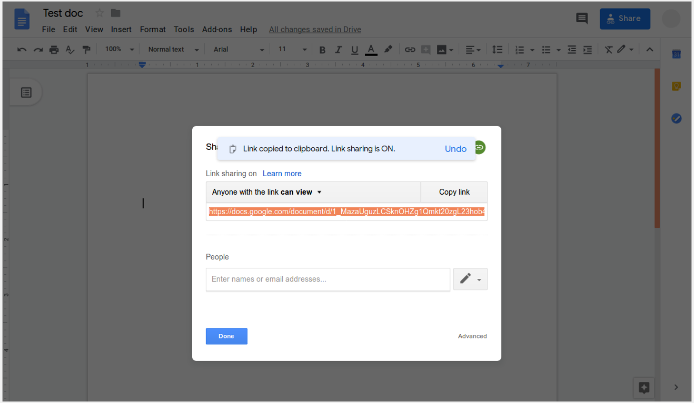


Figure 9. Click the Share button.

- Collaborators can be invited to help work on a document.
- Can add their email address or send them a link to the document.



Link copied to clipboard. Link sharing is ON. Undo

Link sharing on Learn more

Anyone with the link **can view** Copy link

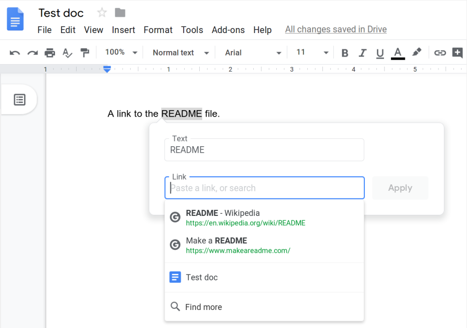
https://docs.google.com/document/d/1_MazaUguzLCSknOHZg1Qmkt20zgL23hob4

People

Enter names or email addresses...

Figure 10. Sharing a Doc.

Linking to a file



A link to the README file.

Text: README

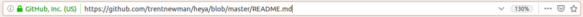
Link: Paste a link, or search

README - Wikipedia
https://en.wikipedia.org/wiki/README

Make a README
https://www.makeareadme.com/

Test doc

Figure 11. A Doc link.

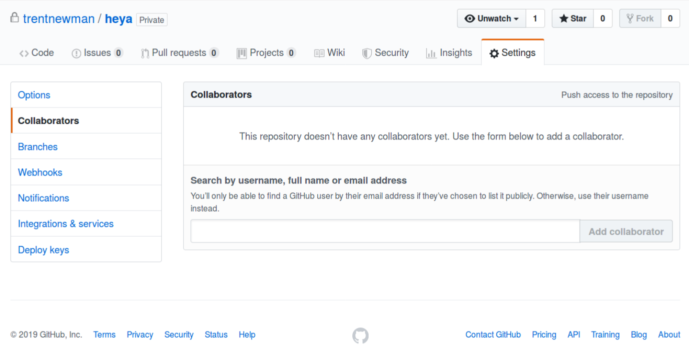


https://github.com/trentnewman/heya/blob/master/README.md

Figure 12. Adress to GitHub file.

- Links to up-to-date files in your GitHub repo can be entered in the Google Doc.
- Select the word (or symbol) that you want to turn into a link, right click and select Link.
- Enter the address to the file of interest in your GitHub repo.
- Note: if you change the location of file in the repo, you will have to update the link address.

Sharing a private repo



Collaborators Push access to the repository

This repository doesn't have any collaborators yet. Use the form below to add a collaborator.

Search by username, full name or email address

You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.

Figure 13. Inviting a collaborator.

- If the link in your Google Doc links to a file in a private repository your collaborator will need access to see it.
- Go the Setting tab for your repo you can invite collaborators to provide them with access to your private repo.

A “minimal” analysis pipeline

- Researchers who write code can more efficiently manage data sets, crunch and clean up the numbers, and visualize the results.
- In our GitHub repo we will create a system that comprises of;
  - An input file.
  - An analysis script.
  - An output file.
- We can then make a link to the output file on GitHub so that our collaborators can see the latest results.
- There are many great free programming languages, but will use Python here as it often comes pre-installed on computers.
  - Python is also “general-use”, compared languages like R and Julia which focus on data science.

Ref: Perkel, Jeffrey M. "Programming: pick up Python." Nature News 518.7537 (2015): 125.

A data input file

- Data entry is typically be done in Microsoft Excel (or the free LibreOffice Calc).
- Though you may often save files in .xlsx form, it is easier to work with exported .csv (comma separated values).
  - Note: Highlighting cells in different colors is not useful for analysis.
  - Different tables should be in different sheets and exported individually.
- Try to arrange data such that each column is a distinct variable.
- The filename of a dataset can often contain valuable information, e.g. immuno_rad51_female_2018-12-01.csv.
  - If you use a consistent symbol to separate this information (e.g. "_"), and name spreadsheets consistently, the information can be extracted incorporated into the analysis.

An example input

- A file of comma-separated values can be created in a text editor (or Excel).

```
cat input.csv
date,a
1,10
1,11
2,12
2,12
```

Check Python installation

- Open a terminal window, and enter the following command.

```
python --version
```

- You should get something like: Python 2.7.12.
- You may also have Python 3 on your computer, check with:

```
python3 --version
```

- While you should use Python3 where possible, Pyhon2 is still in common use.
- To start a Python session in your terminal enter (for Python3):

```
python3
```

- You should get a prompt that starts with >>>.

Specialized modules

- There are many specialized modules for carrying out certain tasks in python.
  - Many of these are actively maintained on GitHub.
- The packages are available from pypi.org and can be installed with the pip command.
- To check if you have pip installed, type the following into the terminal:

```
python -m pip --version
```

- We will need the Pandas module (processes dataframes in a manner similar to R) for working with this input data.

```
pip install pandas #may have to run as sudo
```

A script

- Though you can enter commands one by one in the terminal, it is better write scripts.
- A script is a set of instructions for the computer.
    - An alternative to a script is notebook, like Jupyter notebook, for working with the code interactively.
- Below is a script that will calculate summary statistics for the input data, save as script.py

```
import pandas # Import the pandas module
df = pandas.read_csv('input.csv') # read the input file
out = df.groupby(['date'])['a'].agg(['mean','std','count']) # aggregate statistics for data grouped by date variable
out = out.reset_index() # reset dataframe variable
out.to_csv('output.csv', index=False) # save output as a csv
```

- Once the script is written enter python script.py to run it and produce the output.csv file.

Updating your repo

- With a working pipeline in your repo you can push the changes to GitHub and link Google Doc to the output file.
- Now try adding more data to the input file, rerunning the script, and committing the changes.
- The process of re-running the script, and pushing the changes to GitHub can become tedius and can be automated with a bash script (e.g. run.sh).

```
#!/usr/bin/env bash
python script.py
git add .
git commit -m "reloaded"
git push origin master
```

- Make the script executable by typing chmod +x run.sh Now all you need to do to update your repo is type ./run.sh into the terminal.

Advanced topics

Outlined above is a basic workflow for sharing up-to-date data with collaborators, advanced topics for consideration include:

- Working with large files, using GitHub LFS, and paid data packs.
- Using Git to merging changes to your repo when your local origin is not up-to-date using.
- Producing graphs that can be automatically updated with new data, using ggplot2 (R) or matplotlib (Python).
- Working with notebooks like Jupyter, and converting notebooks into executable ipython scripts.
- Arranging plots into figures using LaTeX, and running with pdflatex.
- Resetting the pipeline so that previous analysis is deleted prior to re-running the script.

Conclusion

Figure 14. Octocat – the GitHub mascot.

- GitHub provides a free platform for sharing data.
- Providing links in a Google Doc can allow collaborators to see the most up-to-date results.
- Scripting repetitive tasks can require a little work upfront but make workflows more efficient.